NASA Armstrong’s Approach to Store Separation Analysis

Chris Acuff
Trong Bui
Flow Physics Group / Code RA
NASA Armstrong Flight Research Center
Edwards, CA 93523
chris.acuff@nasa.gov
Outline

Store Separation Background

– What is Store Separation?
– Why is NASA Armstrong Pursuing This Capability?

Toolset & Methodology Introduction

– Major Approaches in Store Separation Analysis
– Cart3D – Euler CFD Code
– Star-CCM+ - Full Navier-Stokes CFD Code
– Python Scripting Work
– Workflow Overview

Store Separation Analysis Example

– Initialization of Store Separation Capability for Towed Glider Air Launch System
Store Separation Background

What is Store Separation?

• Store – any device meant for external or internal carriage and mounted on aircraft suspension or release equipment
• Store separation – detachment of store from vehicle
• Safe and Acceptable Separation
  – Safe
    • Store does not hit aircraft or other stores
    • Store does not disintegrate or explode
  – Acceptable
    • Store does not tumble
    • Rates are captured if equipped with sensors
    • Photogrammetric data is captured (if technique is used)

Why is NASA Armstrong Pursuing This Capability?

• Support the airworthiness and flight safety review process at Armstrong
• Help ensure safety of vehicle and crew
• Help ensure success of mission
• Support advocacy for new projects
• Support research efforts in-house and by potential customers
• Complement our flight research/test capabilities, as well as our self-certifying status
Store Separation Gone Wrong!

From https://www.youtube.com/watch?v=fPTnmZ_HPAs
Toolset & Methodology
Introduction
Major Approaches in Store Separation Analysis

**NAVSEP**
Essentially a 6-DoF trajectory analysis for the store, given the initial launch/drop conditions. All aerodynamics properties for the store, as well as the carrying aircraft’s influence are interpolated from look-up tables derived from CFD, wind tunnel tests, or flight testing

- **Advantages:**
  - Very fast computer run-time per simulation
  - Required for sensitivity studies/Monte-Carlo analyses where thousands of simulations are required

- **Disadvantages:**
  - Difficult to account for complex, time-accurate interactions between the store and carrying aircraft
  - Trajectory might be off due to interpolation from sparse look-up table

**DIRECT TIME-ACCURATE CFD**
The flow around both the store and the carrying aircraft is fully modeled in a time-accurate and transient manner in the CFD solver

- **Advantages:**
  - No interpolation involved
  - Highest fidelity store separation analysis possible with current analysis techniques
  - Provides verification of the NAVSEP results

- **Disadvantages:**
  - Time-, compute-, and labor-intensive
  - Only tens of simulations are realistically possible in an analysis campaign
NAVSEP Introduction

Summary of NAVSEP

• Generalized store separation code used by the US NAVY
• Developed by the Store Separation Branch at NAVAIR
• Tracks the 6 DoF trajectory of the store using aerodynamic look-up tables
• Can be loosely-coupled to a CFD code
• Capabilities:
  – Can input thrust profile (time vs. thrust)
  – Can input ejector forces (time or displacement vs. force)

Inputs:
• Aerodynamic databases (freestream and grid)
• Mass properties
• Flight condition
• Reference lengths / CG locations
• Initial position and orientation in defined coordinate system

Outputs:
• Velocity
• Position
• Orientation
• Forces
• Moments
• Time
• Orientation rate of change
• Alpha
• Beta
Cart3D Introduction

Summary of Cart3D

• High-fidelity Euler code
  – No boundary layer, no viscous effects, no turbulence models
• Unstructured, adaptively refined Cartesian grids
• Efficient for complex geometries
• Turn around time is fast for a single case compared to a full Navier-Stokes run
  – Simply need a surface grid and input files
• Can use multiple processors to decrease run time
• Developed at the NASA Ames Research Center
• Capable of wide range of Mach numbers
Star-CCM+ CFD Code

- Unstructured polyhedral full Navier-Stokes CFD code with Euler, RANS, DES, and LES approximations
- Unsteady/time-accurate and overset mesh capabilities allow arbitrarily complex geometries to be analyzed in a store separation analysis
- In continuous use at NASA Armstrong since 2008 with good results in support of a wide variety of flight projects including our GIII and F-15 flight research testbeds
- Used together with Cart3D to construct aerodynamic look-up tables for the NAVSEP approach
- Also could be used to conduct direct time-accurate CFD store separation analysis
Python Scripting

Summary of Python Use

• Python is used to set up Cart3D runs for creation of aerodynamic databases
  – Creates folder structure, translates geometry, and creates all input files for Cart3D

• Python is also used to parse NAVSEP output and create plots of store position and orientation over the course of the simulation

• Currently using Anaconda Python Distribution
  – Python distribution that includes many packages that are useful for scientific / engineering work
  – Includes Python
    • Key packages utilized
      – Matplotlib – used for creating plots
      – Spyder (integrated development environment) – useful for writing / debugging code

[Python logo and Anaconda logo]
Assume initial store release conditions

Conduct sensitivity studies using NAVSEP

Finalize the design release condition from NAVSEP sensitivity studies

Verify clearance for the final design release conditions using Star-CCM+

Provide results to support project advocacies or flight readiness reviews

All simulation files and scripts are saved to support future analysis requirements for similar projects/airplanes and/or mishap investigations
NAVSEEP Workflow for Store Separation Analysis

1. Clean up geometry, create surface grids
2. Run AVL / formulate CFD runs
3. Run Cart3D
4. Collect aerodynamic data and create NAVSEEP input files
5. Run NAVSEEP
6. Parse output files / plot relevant data
Star-CCM+ Workflow for Store Separation Analysis

- Clean up geometry, create surface grids
- Create volume grids
- Run Star-CCM+
- Conduct grid and time step refinement studies
- Post-process results and report
Validation of Computational Toolset and Models

How Would We Validate Our Computational Toolset and Models?

• Perform mesh and time step refinements

• Compare to available store separation validation datasets

• Compare to available wind tunnel test data

• Compare to flight data:
  – Photogrammetry data
  – Differential GPS data between store and carrying aircraft
  – Inertial 6-DoF data packs in both store and carrying aircraft
  – A combination of all these approaches to provide a more comprehensive understanding as well as redundancies in data collection
Store Separation Analysis Example
Towed Glider Air Launch System (TGALS)
Aircraft CAD / Clean-Up

Twin fuselage glider used as the aircraft

- High-quality laser scan is available for configuration
- Laser scan was performed by Operations Engineering branch

CAD clean-up was relatively simple

- Several areas had self-intersecting geometry problems (green circle)
- These areas were approximated with 3D splines and adjusted to fix the issues
Aerospike rocket from the Dryden Aerospike Rocket Test Project served as the initial store for analysis purposes

- Used only for store separation demonstration purposes. Final store geometry and mass properties will be used as available
- Aerospike rocket is a great choice for this initial look, since we already have geometry and mass properties data

Store CAD creation

- Rocket dimensions and mass properties taken from publicly-available publications
Surface Grid Generation

Glider and Rocket Surface Grid Generation

- Created triangulation in Pointwise
- Uniform spacing of \( ds = 0.5 \) on both glider and rocket
- Since grid is created for Euler simulation, not much refinement as been performed for leading edges, trailing edges, areas of high curvature, etc...
- Geometry preservation was not great on nosecone, so triangulation was made from a structured grid of the nose cone
- CAD was cleaned-up before grid generation commences
Athena Vortex Lattice Model

An AVL Model was created to obtain damping coefficients required by NAVSEP

- Store roll-damping coefficient, $C_{lp}$
- Store pitch-damping coefficient, $C_{mq}$
- Store yaw-damping coefficient, $C_{nr}$
- Model consists of fins with general outline of body of rocket
- User’s guide suggests excluding body, but it seemed to make a difference in the damping coefficients
What NAVSEP Needs

In order to calculate a store trajectory, NAVSEP needs an aerodynamic database for 2 scenarios:

– Aerodynamic data about just the rocket in the freestream (freestream case)
– Aerodynamic data about the rocket in the influence of the mothership (grid case)
CFD Run Formulation, continued

Freestream CFD Runs

• 7 points were chosen to create the aero databased NAVSEP needs
  – 7 different angles of attack, all at $\beta = 0^\circ$
CFD Run Formulation, continued

Grid CFD Runs

• 9 points were chosen to create the aero database NAVSEP needs
  – 9 different X, Y, Z positions, $\beta = 0^\circ$
Scripts were reused from earlier efforts

• No need to really do much work setting up NAVSEP
• Python scripts work well for running and plotting NAVSEP
• Run time <1 second
  – Takes Python longer to plot the results than it does to run NAVSEP
• All that was needed were the new grid and freestream files created from the Cart3D runs
  – These were created by hand using the results from the Cart3D runs
• Once everything was in place, NAVSEP ran great!
NAVSEP Results: Trajectory
NAVSEF Results: Displacement
NAVSEP Results: Orientation
Store Separation Animation

- Preliminary NAVSEP result only
- Clean separation for the present aircraft and store as well as release conditions
- Final design release condition will be verified using Star-CCM+ Navier-Stokes code before flight